

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

IN RE APPLICATION OF:	§	ATTY. DOCKET NO.:	GB920030079US1
	§		
RICHARD L. KULP, ET AL.	§	EXAMINER:	KIM LYNN DAM
	§		
SERIAL NO.:	§	CONFIRMATION NO.:	5413
10/720,804	§		
	§		
FILED: 24 NOVEMBER 2003	§	ART UNIT:	2179
	§		
FOR: DESIGN TOOL	§		

APPEAL BRIEF UNDER 37 C.F.R. 41.37

Mail Stop Appeal Briefs - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

This Brief is submitted in support of the Appeal of the Examiner's final rejection of Claims 1-21 in the above-identified application. A Notice of Appeal was filed electronically in this case on September 11, 2007 and received in the United States Patent and Trademark Office on September 11, 2007. A three-month extension of time is hereby requested to extend the period for filing this Brief to February 11, 2008. Please charge the fee of \$510.00 due under 37 C.F.R. §1.17(c) for filing the brief, as well as any additional required fees, to **IBM CORPORATION DEPOSIT ACCOUNT No. 09-0461**.

REAL PARTY IN INTEREST

The real party in interest in the present Application is International Business Machines Corporation, the Assignee of the present application.

RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellants, the Appellants' legal representative, or assignee, which directly affect or would be directly affected by or have a bearing on the Board's decision in the pending appeal.

STATUS OF CLAIMS

Claims 1-32 stand finally rejected by the Examiner as noted in the Final Office Action dated July 13, 2007 ("Final Office Action"). The rejection of Claims 1-32 under 35 U.S.C. §103(a) are appealed.

STATUS OF AMENDMENTS

Appellants filed Amendment B under 37 C.F.R. §1.116 on February 11, 2008 subsequent to the July 13, 2007 Final Office Action from which this Appeal is filed. Appellants' understanding is that the Examiner has not acted upon said Amendment B at the time of filing this Appeal Brief.

SUMMARY OF THE CLAIMED SUBJECT MATTER

As set forth in independent **Claim 1**, "a method for performing an operation on a graphic object in a display of overlapping graphic objects in a data processing system" (supported, for example, in the originally filed specification on p. 10, lines 1-6) includes the steps of:

"storing a model of a graphic object, wherein said model includes an indication of whether said graphic object is a parent or child of another graphic object" (supported, for example, on p. 10, lines 6-10; p. 11, lines 3-9; Fig. 2, step 100);

"displaying a plurality of overlapping graphic objects to a user" (supported, for example, at p. 10, lines 6-10; p. 10, lines 25-27; Fig. 3); "detecting a selection by the user of a graphics

editing operation to be performed on a graphic object” (supported, for example, on p. 11, lines 11-26; see also Fig. 3, toolbar 64);

“detecting a position of a pointer with respect to a display of said plurality of overlapping graphic objects” (supported, for example on p. 10, lines 10-15; Fig. 2, step 104);

“displaying to a user a list of overlapping graphic objects which coincide with said pointer position and on which said graphics editing operation can be performed” (supported, for example, on p. 10, lines 13-18; p. 12, lines 9-15; Fig. 2, step 106; Fig. 4, hover window 70);

“detecting a selection by the user of one graphic object of said indicated plurality of overlapping graphic objects as a target graphic object without the user changing said pointer position to make said selection” (supported, for example, on p. 10, lines 18-21; p. 12, lines 15-21; Fig. 2, step 108; Fig. 6, step 108);

“performing said graphics editing operation on said target graphic object” (supported, for example, on p. 10, lines 18-21; p. 12, lines 23-25; Fig. 2, step 110; Fig. 6, steps 110); and

“making said target graphic object visible during performance of the graphics editing operation on said target graphic object” (supported, for example, p.12, line 25 to p. 13, line 3; Fig. 6, step 114).

As set forth in independent **Claim 16**, “a software tool for performing a graphics editing operation on a graphic object in a display of overlapping graphic objects in a data processing system” (supported, for example, in the originally filed specification on p. 20, lines 7-18) the tool being operable to:

“store a model of a graphic object, wherein said model includes an indication of whether said graphic object is a parent or child of another graphic object” (supported, for example, on p. 10, lines 6-10; p. 11, lines 3-9; Fig. 2, step 100);

“display a plurality of overlapping graphic objects to a user” (supported, for example, at p. 10, lines 6-10; p. 10, lines 25-27; Fig. 3);

“detect a selection by the user of a graphics editing operation to be performed on a graphic object” (supported, for example, on p. 11, lines 11-26; see also Fig. 3, toolbar 64);

“detect a position of a pointer with respect to a display of said plurality of overlapping graphic objects” (supported, for example on p. 10, lines 10-15; Fig. 2, step 104);

“display to a user a list of overlapping graphic objects which coincide with said pointer position and on which said graphics editing operation can be performed” (supported, for example, on p. 10, lines 13-18; p. 12, lines 9-15; Fig. 2, step 106; Fig. 4, hover window 70);

“detect a selection by the user of one graphic object of said indicated plurality of overlapping graphic objects as a target graphic object without the user changing said pointer position to make said selection” (supported, for example, on p. 10, lines 18-21; p. 12, lines 15-21; Fig. 2, step 108; Fig. 6, step 108);

“perform the graphics manipulation operation on said target graphic object” (supported, for example, on p. 10, lines 18-21; p. 12, lines 23-25; Fig. 2, step 110; Fig. 6, steps 110); and

“making said target graphic object visible during performance of the graphics editing operation on said target graphic object” (supported, for example, p.12, line 25 to p. 13, line 3; Fig. 6, step 114).

As set forth in independent **Claim 28**, “A program element comprising program code operable to provide the software tool of claim 16” (supported, for example, in the originally filed specification on p. 20, lines 7-18).

As set forth in independent **Claim 30**, “a tangible computer-readable medium comprising a computer program element including computer program instructions to implement the method of claim 1” (supported, for example, in the originally filed specification on p. 9, lines 7-11 and p. 20, lines 20-26).

As set forth in independent **Claim 32**, “a data processing system comprising a software tool according to claim 16” (supported, for example, in the originally filed specification on p. 20, lines 7-18 and Fig. 1).

GROUND S OF REJECTION TO BE REVIEWED ON APPEAL

- A. The Examiner’s rejection of Claims 1-3, 5-10, 15-17, 19-24 and 28-32 under 35 U.S.C. §103(a) as being unpatentable over U.S. Pat. No. 5,574,177 to *Hama et al.* (“*Hama*”) in

view of U.S. Pat. No. 5,627,959 to *Brown et al.* (“*Brown*”) and in further view of Appellants’ own admitted prior art (“Appellants’ APA”) is to be reviewed on Appeal.

- B. The Examiner’s rejection of Claims 4 and 18 under 35 U.S.C. §103(a) as being unpatentable over U.S. Pat. No. 5,574,177 to *Hama et al.* (“*Hama*”) in view of U.S. Pat. No. 5,627,959 to *Brown et al.* (“*Brown*”), in further view of Appellants’ own admitted prior art (“Appellants’ APA”) and in further view of U.S. Pat. No 6,335,733 to *Keren et al.* (“*Keren*”) is to be reviewed on Appeal.
- C. The Examiner’s rejection of Claims 11 and 25 under 35 U.S.C. §103(a) as being unpatentable over U.S. Pat. No. 5,574,177 to *Hama et al.* (“*Hama*”) in view of U.S. Pat. No. 5,627,959 to *Brown et al.* (“*Brown*”), in further view of Appellants’ own admitted prior art (“Appellants’ APA”) and in further view of U.S. Pat. No 5,377,314 to *Bates et al.* (“*Bates*”) is to be reviewed on Appeal.
- D. The Examiner’s rejection of Claims 12-14 and 26-27 under 35 U.S.C. §103(a) as being unpatentable over U.S. Pat. No. 5,574,177 to *Hama et al.* (“*Hama*”) in view of U.S. Pat. No. 5,627,959 to *Brown et al.* (“*Brown*”), in further view of Appellants’ own admitted prior art (“Appellants’ APA”), in further view of U.S. Pat. No 5,377,314 to *Bates et al.* (“*Bates*”) and in further view of U.S. Pat No. 5,651,107 to *Frank et al.* (“*Frank*”) is to be reviewed on Appeal.

ARGUMENTS

- A. The Examiner’s rejection of Claims 1-3, 5-10, 15-17, 19-24 and 28-32 under 35 U.S.C. §103(a) as being unpatentable over *Hama* in view of *Brown* and in further view of Appellants’ APA is to be reviewed on Appeal.

1. The Examiner’s rejection of Claims 1-3, 5-10, 15-17, 19-24 and 28-32 is improper because the combination of *Hama*, *Brown* and Appellants’ APA does not render obvious the step of “displaying to a user a list of overlapping graphic objects

which coincide with said pointer position and on which said graphics editing operation can be performed”, as claimed.

Regarding the rejection of exemplary Claim 1, the combination of *Hama*, *Brown* and Appellants’ APA does not render obvious the step of “displaying to a user a list of overlapping - graphic objects which coincide with said pointer position and on which said graphics editing operation can be performed”, as claimed. The Examiner cites *Hama* as teaching the claimed feature, stating that the “editing operation is inherently performed on the selected edit object” (Final Office Action, p. 5, paragraph 7). However, the clause “on which said graphics editing operation can be performed” describes the list of overlapping graphic objects that are displayed to the user. For example, if the user selects a paint operation, then the list displayed to the user will only display overlapping graphics objects that can be painted. Overlapping graphics objects that cannot be painted will not be displayed in the list. There is nothing “inherent” about this functionality, as the Examiner suggests. The *Hama* reference teaches, “the present invention selects any and all edit objects which have even a part thereof contained in the set specified region as selective candidates” (col. 4, lines 19-24). *Hama* also teaches displaying “a subscreen in which information relating to all selective candidates is displayed” (col. 3, line 66 to col. 4, line 1). The subscreen (or list) of *Hama* displays all selective candidates (selected graphics objects). The subscreen of *Hama* does not display those overlapping graphics object on which said graphics editing operation can be performed, as claimed.

The Examiner cites no evidence in either the *Brown* reference or in Appellants’ APA as teaching, “displaying to a user a list of overlapping graphic objects which coincide with said pointer position and on which said graphics editing operation can be performed”, as claimed. Therefore, the combination of *Hama*, *Brown* and Appellants’ APA does not render obvious exemplary Claim 1.

2. The Examiner’s rejection of Claims 1-3, 5-10, 15-17, 19-24 and 28-32 is improper because the combination of *Hama*, *Brown* and Appellants’ APA does not render obvious the step of “detecting a selection by the user of one graphic object of

said indicated plurality of overlapping graphic objects as a target graphic object without the user changing said pointer position to make said selection”, as claimed.

Regarding the rejection of exemplary Claim 1, the combination of *Hama*, *Brown* and Appellants’ APA does not render obvious the step of “detecting a selection by the user of one graphic object of said indicated plurality of overlapping graphic objects as a target graphic object without the user changing said pointer position to make said selection”, as claimed. The Examiner states:

Hama does not specifically teach detecting a selection by the user of one graphic object of said indicated plurality of overlapping graphic objects as a target without the user changing said pointer position to make said selection. However, selection of an object from a menu or subscreen without the user changing said pointer position, for example keyboard arrow buttons to move along choices and enter button to select, is well known and expedient in the art. (Final Office Action, p. 5, paragraph 7).

The Examiner asserts that the selection of an object from a menu without the user changing said pointer position is well known and expedient in the art. However, *Hama* teaches selecting “the desired edit object by suitably moving the pointer within the subscreen” (col. 5, lines 4-6). *Hama* teaches, “the operator sequentially changes the display state of the edit object by suitably moving the pointer” (col. 4, lines 66-67). *Hama* teaches, “the figure corresponding to the information region 75 which is pointed to by the pointer is highlighted by moving the pointer on the subscreen so that the desired figure can be easily selected” (col. 7, lines 14-17). *Hama* also teaches, “the present invention changes the display of a corresponding edit object in the main screen which is the entire screen by moving the pointer in the zoomed screen” (col. 7, lines 36-38). Clearly, *Hama* relies on moving a pointer to make selections of objects, which teaches away from Appellant’s claimed invention. Therefore, it would not be obvious to one skilled in the art at the time of the invention to combine making selections without changing the position of a pointer (which the Examiner asserts is well-known in the art) with *Hama*, which teaches moving a pointer to select graphic objects.

The Examiner cites no evidence in either the *Brown* reference or in Appellants’ APA as teaching, “detecting a selection by the user of one graphic object of said indicated plurality of overlapping graphic objects as a target graphic object without the user changing said pointer

position to make said selection”, as claimed. Therefore, the combination of *Hama*, *Brown* and Appellants’ APA does not render obvious exemplary Claim 1.

The rejection of independent Claims 16, 28, 30 and 32 is improper for the reasons adduced above regarding the rejection of exemplary Claim 1. Claims 2-3, 5-10, 15, 17, 19-24, 29 and 31 depend directly or indirectly from independent Claims 16, 28, 30 and 32, which have been improperly rejected in view of the cited art. The rejection of Claims 2-3, 5-10, 15, 17, 19-24, 29 and 31 is likewise improper and should be reversed in view of the arguments put forth herein regarding the rejection of independent Claims 16, 28, 30 and 32.

B. The Examiner’s rejection of Claims 4 and 18 under 35 U.S.C. §103(a) as being unpatentable over *Hama* in view of *Brown*, in further view of Appellants’ APA and in further view of *Keren* is to be reviewed on Appeal.

Claims 4 and 18 depend directly from independent Claims 1 and 16, which have been improperly rejected in view of the cited art. The rejection of Claims 4 and 18 is likewise improper and should be reversed in view of the arguments put forth herein regarding the rejection of independent Claims 1 and 16.

C. The Examiner’s rejection of Claims 11 and 25 under 35 USC 103(a) as being unpatentable over *Hama* in view of *Brown*, in further view of Appellants’ APA and in further view of *Bates* is to be reviewed on Appeal.

Claims 11 and 25 depend directly from independent Claims 1 and 16, which have been improperly rejected in view of the cited art. The rejection of Claims 11 and 25 is likewise improper and should be reversed in view of the arguments put forth herein regarding the rejection of independent Claims 1 and 16.


D. The Examiner’s rejection of Claims 12-14 and 26-27 under 35 U.S.C. §103(a) as being unpatentable over *Hama* in view of *Brown*, in further view of Appellants’ APA, in further *Bates* and in further view of *Frank* is to be reviewed on Appeal.

Claims 12-14 and 26-27 depend directly from independent Claims 1 and 16, which have been improperly rejected in view of the cited art. The rejection of Claims 12-14 and 26-27 is likewise improper and should be reversed in view of the arguments put forth herein regarding the rejection of independent Claims 1 and 16.

CONCLUSION

Appellants have pointed out with specificity the manifest error in the Examiner's rejections, and the claim language which renders the invention patentable over the various combinations of references. Appellants, therefore, respectfully request the reversal of the rejections of all pending claims.

Respectfully submitted,



James E. Boice
Reg. No. 44,545
DILLON & YUDELL LLP
8911 N. Capital of Texas Highway
Suite 2110
Austin, Texas 78759
512-343-6116

ATTORNEY FOR APPELLANTS

CLAIMS APPENDIX

1. A method for performing an operation on a graphic object in a display of overlapping graphic objects in a data processing system, the method comprising the steps of:
 - storing a model of a graphic object, wherein said model includes an indication of whether said graphic object is a parent or child of another graphic object;
 - displaying a plurality of overlapping graphic objects to a user;
 - detecting a selection by the user of a graphics editing operation to be performed on a graphic object;
 - detecting a position of a pointer with respect to a display of said plurality of overlapping graphic objects;
 - displaying to a user a list of overlapping graphic objects which coincide with said pointer position and on which said graphics editing operation can be performed;
 - detecting a selection by the user of one graphic object of said indicated plurality of overlapping graphic objects as a target graphic object without the user changing said pointer position to make said selection;
 - performing said graphics editing operation on said target graphic object; and
 - making said target graphic object visible during performance of the graphics editing operation on said target graphic object.
2. The method of claim 1, wherein said position of a pointer comprises a user-defined area of said display described by the motion of said pointer in response to said user dragging an input device.
3. The method of claim 1, wherein said step of selecting the graphics editing operation comprises detecting a user input identifying selection of a graphics editing tool.
4. The method of claim 3, further comprising the step of changing the form of said pointer on said display in response to detection of said user input selection

5. The method of claim 1, wherein said step of selecting the graphics editing operation to be performed comprises selecting a source graphic object and said step of performing the graphics editing operation comprises dropping said source graphic object onto said target graphic object.
6. The method of claim 1, further comprising the step of storing the positions of said plurality of overlapping graphic objects on said display and comparing said pointer position with said graphic object positions to determine which graphic objects are coincident with the pointer.
7. The method of claim 1, further comprising the step of maintaining a record of attributes relating to each of said graphic objects.
8. The method of claim 1, wherein said displaying step comprises continuously displaying to the user a hover window listing said graphic objects which are coincident with said pointer position and continuously updating said hover window in response to changes in said pointer position.
9. The method of claim 1, wherein said graphics editing operation comprises adding text to said target object.
10. The method of claim 9, wherein said graphics editing operation further comprises the steps of:
 - opening a text box on said target graphic object;
 - displaying a text insertion cursor in said text box to display the location where new text will be inserted; and
 - ending said graphic operation in response to the user moving said pointer to a position outside of said text box and depressing a control button on an input device.
11. The method of claim 1, further comprising the step of determining whether said target graphic object is the outermost one of said graphic objects which are coincident with the position of the pointer.

12. The method of claim 11, wherein the step of making said target graphic object visible comprises temporarily making at least the outermost one of said coincident graphic objects transparent.

13. The method of claim 12, wherein the outermost coincident graphic object is a child graphic object to a parent graphic object and the step of making the selected target graphic object visible comprises temporarily making said parent graphic object transparent in response to making said child graphic object transparent.

14. The method of claim 12, wherein said outermost one of the coincident graphic objects reappears automatically after the graphics editing operation has been performed on said target graphic object.

15. The method of claim 1, wherein said step of displaying a list of overlapping graphic objects is dependent on said position of the pointer remaining the same for a certain period of time.

16. A software tool for performing a graphics editing operation on a graphic object in a display of overlapping graphic objects in a data processing system, the tool being operable to:

- store a model of a graphic object, wherein said model includes an indication of whether said graphic object is a parent or child of another graphic object;

- display a plurality of overlapping graphic objects to a user;

- detect a selection by the user of a graphics editing operation to be performed on a graphic object;

- detect a position of a pointer with respect to a display of said plurality of overlapping graphic objects;

- display to a user a list of overlapping graphic objects which coincide with said pointer position and on which said graphics editing operation can be performed;

- detect a selection by the user of one graphic object of said indicated plurality of overlapping graphic objects as a target graphic object without the user changing said pointer position to make said selection;

- perform the graphics manipulation operation on said target graphic object; and

making said target graphic object visible during performance of the graphics editing operation on said target graphic object.

17. The software tool of claim 16, wherein said position of a pointer comprises a user-defined area of said display described by the motion of said pointer in response to said user dragging an input device.

18. The software tool of claim 16, operable to change the form of said pointer on said display in response to detection of said user input selection.

19. The software tool of claim 16, operable to select a source object and drop said source object onto said target graphic object.

20. The software tool of claim 16, operable to store the positions of said plurality of overlapping graphic objects on said display and compare said pointer position with said graphic object positions to determine which graphic objects are coincident with the pointer.

21. The software tool of claim 16, operable to maintain a record of attributes relating to each of said graphic objects.

22. The software tool of claim 16, operable to continuously display to the user a hover window listing said graphic objects which are coincident with said pointer position and continuously updating said hover window in response to changes in said pointer position.

23. The software tool of claim 16, editing operation comprises adding text to said target object.

24. The software tool of claim 23, wherein said graphics editing operation further comprises the steps of:

opening a text box on said target graphic object;

displaying a text insertion cursor in said text box to display the location where new text will be inserted; and

ending said graphic operation in response to the user moving said pointer to a position outside of said text box and depressing a control button on an input device.

25. The software tool of claim 16, operable to determine whether said target graphic object is the outermost one of said graphic objects which are coincident with the position of the pointer.

26. The software tool of claim 24, operable to make at least the outermost one of said coincident graphic objects temporarily transparent.

27. The software tool of claim 26, operable to make said outermost one of the coincident graphic objects reappear automatically after the graphics editing operation has been performed on said target graphic object.

28. A program element comprising program code operable to provide the software tool of claim 16.

29. The program element of claim 28 on a tangible, computer-readable medium.

30. A tangible computer-readable medium comprising a computer program element including computer program instructions to implement the method of claim 1.

31. The tangible computer-readable medium of claim 30, comprising one or more of the following set of media: a signal, a magnetic disk or tape, solid-state memory, a compact disk and a digital versatile disk.

32. A data processing system comprising a software tool according to claim 16.

EVIDENCE APPENDIX

Other than the Office Action(s) and reply(ies) already of record, no additional evidence has been entered by Appellants or the Examiner in the above-identified application which is relevant to this appeal.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings as described by 37 C.F.R. §41.37(c)(1)(x) known to Appellants, Appellants' legal representative, or assignee.